Equation Engine – SF Amount Equations and Adjustment Calendars

Session #26498
March 24, 2009

Alliance 2009 Conference
Anaheim, CA
Your Presenter

Aaron Apel
University of Wisconsin – Madison

- 6 years experience with PeopleSoft with a focus on Student Financials
- Configure and administer the Tuition Assessment side of SF
This presentation will focus primarily on the setup and configuration of the delivered Equation Engine functionality used to generate accurate tuition amounts in conjunction with the adjustment calendars to account for enrollment add/drop activity.
Agenda

- General SF Equations overview
- SF Amount Equations
- SF Amount Equations used with Adjustment Calendars
University of Wisconsin - Madison

- UW-Madison enrolls over 40,000 students annually in over 400 programs across campus.

- In Tuition Assessment, we maintain 15 active Tuition Groups and differentiate tuition across 8 Academic Careers.
Oracle PeopleSoft Info

- PeopleSoft v9.0
- Tools Release 8.49.15
- Patched through Bundle 11
SF Equations
Overview

- **SF Select Equations**
  - *Used to identify a specific population of students based on programmed criteria*
  - *Can be used in place of Term Fee Criteria to identify a group to be charged a certain term fee*
  - *More flexible than delivered Term Fee Criteria in that you can program the Equations using ‘and/or’ logic instead of strictly ‘and’*
  - *Ability to use any Record.Field in determining selection criteria given the appropriate security is in place*
SF Equations Overview cont’d

- **SF Amount Equations**
  - *Used to calculate an amount to charge for the term fee the equation is attached to*
  - *Any Record.Field can be used to determine an amount. For example:*
    - $X per course based on program or plan
    - $X per credit based on acad_career
    - $X flat amount as an add-on based on service indicator
SF Equations Overview cont’d

- SF Amount Equations cont’d
  - Can now be used in conjunction with Adjustment Calendars to calculate an accurate tuition amount based on add/drop activity of the student. Fix was delivered with V8.9 bundle #9
SF Amount Equations can be programmed to charge based on criteria built into the equation itself.

The following screenshot shows an example of a simple SF Amount Equation used to charge a flat rate of $100 to any Graduate student who has registered for the term being calc’d.

This could easily be accomplished using either standard Criteria or a SF Select Equation.

IF >

STDNT_CAR_TERM.ACAD_CAREER = 'GRAD'
AND STDNT_CAR_TERM.REGISTERED = 'Y'

THEN CHARGE $100 Add-on
The next screenshot will expand on the criteria from the previous equation. Now the student must be a registered Graduate student, but must also have a value entered in either STDNT_EQUTN_VAR.VARIABLE_CHAR1 or STDNT_EQUTN_VAR.VARIABLE_CHAR2.

The OR statement used on the STDNT_EQUTN_VAR record is not supported by standard Criteria. However, this could still be accomplished using a SF Select equation.

```
IF >

STDNT_CAR_TERM.ACAD_CAREER = 'GRAD'
AND STDNT_CAR_TERM.REGISTERED = 'Y'
AND (STDNT_EQUTN_VAR.VARIABLE_CHAR1 = 'TA'
OR STDNT_EQUTN_VAR.VARIABLE_CHAR2 = 'RA')

THEN CHARGE $100 Add-on
```
<table>
<thead>
<tr>
<th>Line</th>
<th>Component</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>=</td>
<td>String</td>
<td>GRAD</td>
</tr>
<tr>
<td>19</td>
<td>And</td>
<td>Table</td>
<td>STDNT_CAR_TERM</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>Field</td>
<td>REGISTERED</td>
</tr>
<tr>
<td>21</td>
<td>=</td>
<td>String</td>
<td>Y</td>
</tr>
<tr>
<td>22</td>
<td>Then</td>
<td>Table</td>
<td>STDNT_EQUVT_VAR</td>
</tr>
<tr>
<td>23</td>
<td></td>
<td>Field</td>
<td>VARIABLE_CHAR1</td>
</tr>
<tr>
<td>24</td>
<td></td>
<td>String</td>
<td>TA</td>
</tr>
<tr>
<td>25</td>
<td>If</td>
<td>Table</td>
<td>STDNT_EQUVT_VAR</td>
</tr>
<tr>
<td>26</td>
<td></td>
<td>Field</td>
<td>VARIABLE_CHAR2</td>
</tr>
<tr>
<td>27</td>
<td>Or</td>
<td>String</td>
<td>RA</td>
</tr>
<tr>
<td>28</td>
<td></td>
<td>Global Variable</td>
<td>A_AMOUNT</td>
</tr>
<tr>
<td>29</td>
<td>Then</td>
<td>Number</td>
<td>10000000</td>
</tr>
<tr>
<td>30</td>
<td></td>
<td>End Assign</td>
<td></td>
</tr>
</tbody>
</table>
The following screenshot will expand even further on the criteria from the previous examples. It takes advantage of functionality available only through SF Amount equations. Using SF Amount equations, you can charge variable amounts through one Term Fee based on criteria defined within the SF Amount equation attached to that Term Fee.

```
IF >
   STDNT_CAR_TERM.ACAD_CAREER = 'GRAD'
   AND STDNT_CAR_TERM.REGISTERED = 'Y'
   IF >
      STDNT_EQUTN_VAR.VARIABLE_CHAR1 = 'TA'
      THEN CHARGE $100 Add-on
   ELSE IF >
      STDNT_EQUTN_VAR.VARIABLE_CHAR2 = 'RA'
      THEN CHARGE $200 Add-on
```
The true power of SF Amount equations becomes apparent when you utilize the new delivered functionality to charge varying amounts per credit based on enrollment, run the charges through a single Term Fee, and use the adjustment calendars to account for add/drop activity.

For instance, at UW-Madison, we charge Segregated Fees (student use fees) to all students in all careers except for guest students. The individual Seg Fees are charged to their respective Item Types. However, Seg Fees are not charged at the same rate for all students. The rates differ based various criteria; i.e. Academic Career, Student Group, Credit Load, etc.
To accomplish this without equations, our system is structured in such a way that we have an individual Term Fee for each different rate charged for each Segregated Fee. Well over 80 Term Fees to update every term.

Without equations, the Term Fee structure in PeopleSoft limits the flexibility of charging these fees. With equations, we can now reduce those 80+ term fees down to 9.

Here we go....
PS has delivered a set of SF Amount Equation templates that you can use as a basic structure for developing equations to be used in conjunction with adjustment calendars based on several different adjustment models.

SFTDTFSESCLS – Charge by session, adjust by class
SFTDTFSESTRM – Charge by session, adjust by term/session
SFTDTFTRMCLS – Charge by term, adjust by class
SFTDTFTRMSES – Charge by term, adjust by session
SFTDTFTRMTRM – Charge by term, adjust by term

UW Madison uses the charge by term, adjust by session model.
Equation SFTDTFTRMSES is used in the following walkthrough. This equation is 182 lines long just for the basic template.

The equation begins by finding a STDNT_CAR_TERM record for the student being calc’d when the equation is called. It then checks to find any data in STDNT_ENRL for that student. If found, the equation enters a loop to read all enrollment rows that meet the criteria defined within the equation and then calculates the A_AMOUNT to be charged based on billing units; taking into account any adjustments to be made for drop/withdrawal activity.
Enrollment Criteria:

**IF**

\[
\text{stdnt_enrl.stdnt_enrl_status} = 'E' \\
\text{and stdnt_enrl.enrl_add_dt} \leq \text{TRANS\_DATE} \\
\text{and (stdnt_car_term.last_date_attended} = '') \\
\text{or TRANS\_DATE} < \text{stdnt_car_term.last_date_attended})
\]

**OR**

\[
\text{stdnt_enrl.enrl_add_dt} \leq \text{TRANS\_DATE} \\
\text{and stdnt_enrl.enrl_drop_date} > \text{TRANS\_DATE}
\]

**OR**

\[
\text{stdnt_enrl.enrl_add_dt} \leq \text{TRANS\_DATE} \\
\text{and stdnt_enrl.enrl_drop_dt} = \text{TRANS\_DATE} \\
\text{and ENRL\_ACTN\_REASON\_LAST} <> \text{stdnt_car_term.withdraw_reason} \\
\text{and ENRL\_ACTN\_REASON\_LAST} < \text{stdnt_enrl.enrl_actn_reason_last}
\]

**OR**

\[
\text{TRANS\_DATE} = \text{stdnt_enrl.enrl_drop_date} \\
\text{and ENRL\_ACTN\_RSN\_LAST} = \text{stdnt_car_term.withdraw_reason} \\
\text{and SESSION\_CODE} < \text{stdnt_enrl.session_code}
\]
SF Amount Equations w/ Adjustment Calendars

**OR** (TRANS_DATE = stdnt_enrl.enrl_drop_dt and ENRL_ACTN_RSN_LAST <> stdnt_car_term.withdraw_reason and ENRL_ACTN_RSN_LAST = stdnt_enrl.enrl_actn_rsn_last and TRANS_TIME < stdnt_enrl.last_drop_tm_stmp)

**OR** (TRANS_DATE = stdnt_enrl.enrl_drop_dt and ENRL_ACTN_RSN_LAST <> stdnt_car_term.withdraw_reason and ENRL_ACTN_RSN_LAST = stdnt_enrl.enrl_actn_rsn_last and TRANS_TIME = stdnt_enrl.last_drop_tm_stmp and SESSION_CODE < stdnt_enrl.session_code)

**THEN** Calculate charges per credit or flat amount per class and accumulate charges for each class in global variable A_AMOUNT
We were able to expand on the basic structure of this equation to build a set of equations used to calculate the Segregated Fees mentioned earlier.

Instead of accumulating the total A_AMOUNT, the equation was modified to accumulate the total UNT_BILLING for all classes. The total UNT_BILLING was then used to calculate the A_AMOUNT based on extensive criteria built into the equation.

Now, instead of copying and updating the 80+ term fees each term, we insert a new effdt row in the equation and update the amounts charged within the equations.
SF Amount Equations w/ Adjustment
Calendars

IF acad_career = UGST
   return (no charge for Guest Students)

IF acad_career = PHAR
   and acad_level = P4
   assign full-time rate to A_AMOUNT

IF student is Med Full-Time (determined by called equation)
   assign full-time rate to A_AMOUNT

IF acad_career = UGRD, USPC, MEDS, LAW, VMED, PHAR
   IF TOTAL_BILLING_UNITS < 12
      calculate part-time rate and assign to A_AMOUNT
   ELSE-IF TOTAL_BILLING_UNITS > 12
      assign full-time rate to A_AMOUNT
   ELSE-IF TOTAL_BILLING_UNITS > 18
      calculate overload and assign to A_AMOUNT

IF student is in Evening MBA Program (determined by called equation)
   assign EMBA rate to A_AMOUNT
SF Amount Equations w/ Adjustment Calendars

Continued...

**IF** form_of_study = DISR
and TOTAL_BILLING_UNITS <= 3
assign Dissertator rate to A_AMOUNT

**IF** acad_career = GRAD
or acad_prog_primary = MPH
**IF** TOTAL_BILLING_UNITS < 8
calculate part-time rate and assign to A_AMOUNT
**ELSE-IF** TOTAL_BILLING_UNITS > 8
assign full-time rate to A_AMOUNT
SF Amount Equations w/ Adjustment Calendars

UW Madison has a fairly complicated tuition structure including flat-fee ranges, full-time rates, part-time rates, program based rates, and student group based rates. We were able to accommodate all of these with equations.

You can download a PDF copy of the equation outlined above on the HEUG website.
SF Amount Equations are powerful and open the door to some unique possibilities with Tuition Calc, but the flexibility does not come without a cost. Some drawbacks include:

- Somewhat ‘clunky’ programming structure
  - 3 rows to assign a single variable
  - limited to display 100 rows (can collapse rows in v9.0)

- Complicated equations drastically impact Batch Tuition Calc processing speed
  - the equation referenced in this presentation reduced batch tuition calc throughput from approx. 2700 accounts calc’d per hour down to approx. 625 per hour
The reduction in processing efficiency at this time will unfortunately prohibit the implementation of these equations in our production environment.

Not to say that the investigation was futile. Far from it actually. We now have a better understanding of what SF Amount Equations could offer in terms of flexibility. We fully intend to utilize SF Amount Equations in our Tuition Calc structure. However, until the processing impact is addressed, the complexity of the equations used will need to be limited.
Thank you!
Contact Information

- Aaron Apel
- Info Sys Business Automation Senior
- University of Wisconsin – Madison
- Office of the Registrar
- E-mail: adapel@em.wisc.edu
This presentation and all Alliance 2009 presentations are available for download from the Conference Site
Prizes!!!
Complete Evaluation to Win!

- 8 iPod Shuffles
  (one with 4GB given away at Wed a.m. Board Meeting)
- Daily Drawing for the winners
- Just Complete the Online Evaluation
- http://www.heug.org/p/su/rd/sid=283

Complete Evaluation to Win!